Efficient Algorithm for Rectangular Spiral Search

The search pattern is automatically expanded as needed.

NASA's Jet Propulsion Laboratory, Pasadena, California

An algorithm generates grid coordinates for a computationally efficient spiral search pattern covering an uncertain rectangular area spanned by a coordinate grid. The algorithm does not require that the grid be fixed; the algorithm can search indefinitely, expanding the grid and spiral, as needed, until the target of the search is found. The algorithm also does not require memory of coordinates of previous points on the spiral to generate the current point on the spiral.

The search is started at a point (more precisely, in a grid cell), denoted the center of the spiral, that has been chosen previously as the point most likely to coincide with the target. The search is to be performed on a grid of $n \times m$ cells, where $m>n$ and $a \equiv m - n$ is denoted the rectangular excess of the search pattern (see Figure 1). The spiral search is performed in steps numbered simply 1, 2, 3..., and *t* represents the number of the current step.

The inputs to the algorithm are *t* and the rectangular excess that is either specified in advance or calculated from the rectangular grid used to span the initial search area. The output of the algorithm is the pair of integer coordinates (i,j) of the current point on the spiral with respect to the center of the spiral. Figure 2 presents, as an example, the first 15 steps of the spiral generated by this algorithm for a search that starts at a point in a 3×5 rectangle.

This work was done by Paul Brugarolas and William Breckenridge of Caltech for NASA's Jet Propulsion Laboratory.

The software used in this innovation is available for commercial licensing. Please contact Karina Edmonds of the California Institute of Technology at (626) 395-2322. Refer to NPO-42057.

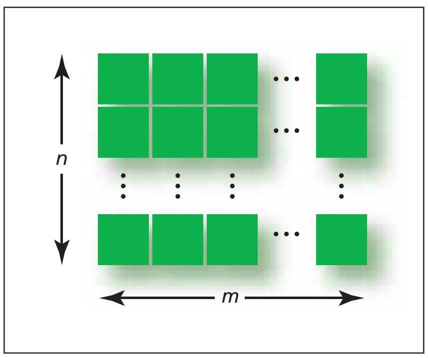

Figure 1. A **Rectangular Grid** of $n \times m$ cells is overlaid on the area to be searched.

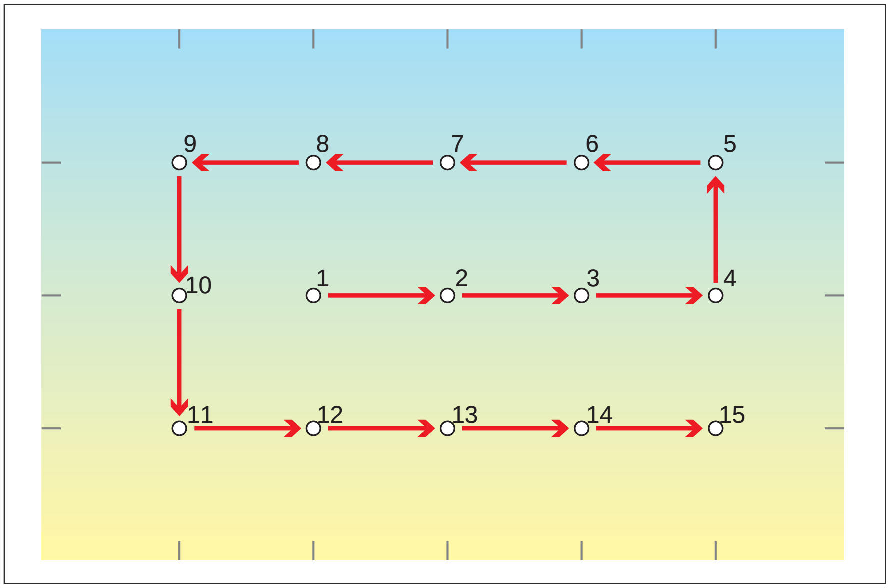

Figure 2. A **Spiral Search Pattern** was generated in an area initially overlaid with a 3×5 grid.

Algorithm-Based Fault Tolerance Integrated With Replication

NASA's Jet Propulsion Laboratory, Pasadena, California

In a proposed approach to programming and utilization of commercial off-the-shelf computing equipment, a combination of algorithm-based fault tolerance (ABFT) and replication would be utilized to obtain high degrees of fault tolerance without incurring excessive costs. The basic idea of the proposed approach is to integrate ABFT with replication such that the algorithmic portions of computations would be protected by ABFT, and the logical portions by replication.

ABFT is an extremely efficient, inexpensive, high-coverage technique for detecting and mitigating faults in computer systems used for algorithmic computations, but does not protect against errors in logical operations surrounding algorithms. Replication is a generally applicable, high-coverage technique for protecting general computations from faults, but is inefficient and costly because it requires additional computation time or additional computational circuitry (and, hence, additional mass and power). The goal of the proposed integration of ABFT with replication is to optimize the fault-tolerance aspect of the design of a computing system by using the less-efficient, more-expensive technique to protect only those computations that cannot be protected